Integration of Product Ontologies for B2B Marketplaces: A Preview

Borys Omelayenko[*]

B2B electronic marketplaces bring together many online suppliers and buyers. Each individual participant potentially can use his own format to represent the products in his product catalog, and these catalogs must be integrated together. Complicated products require knowledge-intensive descriptions, or ontologies, and catalog integration requires integration of product ontologies. The paper surveys the requirements for the integration listed by the industries and current state of the art in ontology integration tools. The survey creates a rough picture of the functionality of the future integration tools as required by the industries.

Keywords: B2B, catalog integration, ontology integration

1. INTRODUCTION

B2B electronic marketplaces bring together many online suppliers and buyers. Each individual participant can potentially use his own format to represent the products in his product catalog. Complicated products require knowledge-intensive descriptions, or ontologies. As a result, catalog integration requires integration of product ontologies. If a marketplace mediates between n suppliers and m buyers, then it must be able to map each of the n suppliers' catalogs into m buyers' formats performing nxm mappings. The numbers n and m can be high enough to make the problem of creation and maintenance of the integration rules non-trivial.

Inference mechanisms developed by the knowledge engineering community provide a standard way to integrate several ontologies. However, the catalogs

---

Address:
[*] Division of Mathematics and Computer Science, Vrije Universiteit Amsterdam, The Netherlands, E-mail: borys@cs.vu.nl, URL: www.cs.vu.nl/~borys

contain a huge number of product descriptions, and this makes inference no longer efficient as a method for catalog integration.

B2B participants mostly rely on XML to define the structure of their product ontologies, and several XML-based standards for product descriptions have been already proposed [4]. Different developers and market players define the ontologies in the terms of different specification languages, they use different naming and language conventions. These differences make the integration problem more complicated.

In Section 2 of this paper we consider the integration task and the main problems that arise during the B2B product integration; in Section 3 we survey the industrial requirements for ontology integration. Section 4 shows state of the art in the ontology integration methods and tools; and we finish the paper with conclusions.

2. INTEGRATION OF PRODUCT ONTOLOGIES

Different suppliers of the B2B marketplaces tend to use different description standards to represent their products on the Web. Luckily, most of the suppliers provide the descriptions in a syntactically unified way (in XML) [4]. However, several important differences occur, which can be classified into syntactic and semantic differences.

The following example shows a syntactical difference in a possible representation of a printer model and its resolution:

```
<printer>
<name>HP LJ 2100</name>
<resolution>1200 dpi</resolution>
<price>699USD</price>
</printer>
```

(a)

```
<printer>
<name>HP LJ</name>
<model>2100</model>
<resolution units="dpi">1200</resolution>
<price>699USD</price>
</printer>
```

(b)

Fig. 1. Syntactical differences in XML encoding

In Fig. 1(a) the printer model is encoded with a single XML tag while in Fig. 1(b) it is encoded with a pair of XML tags. Also, the examples differ in the way in which the printer resolution is represented: example (a) includes the resolution measure (dpi) into the tag resolution, while example (b) uses an additional XML attribute to encode the measure. These differences can be easily fixed by a syntactical translation between the representations.

Another problem occurs because one representation can be semantically richer than another, for example the price can be listed in several currencies, and the element price can be partitioned into amount and currency. Each pair (amount, currency) corresponds to a new currency type and appears once per currency type, as shown in Fig. 2.

```
<price>699USD</price>
```

(a)

```
<price>  <amount>750</amount>
         <currency>DM</currency> </price>
<price>  <amount>699</amount>
         <currency>USD</currency> </price>
```

(b)

Fig. 2. Different expressiveness of the representations

The tags with the same meanings can be named with synonymous names in the same language, or even in different languages (Fig. 3).

```
<price>
    <sum>990</sum>
    <currency>DM</currency>
    <payment>transfer</payment>
</price>
```

(a)

```
<prijs>
<bedrag>990</bedrag>
<valuta>DM</valuta>
<afbetaling>transfer</afbetaling>
</prijs>
```

(b)

Fig. 3. Different languages for XML tags

In Fig. 3(a) the tags are named in English, while in Fig. 3(b) they are named in Dutch, while in both cases they are semantically equal. Even in the same language we can use several synonyms to name the tags, i.e. each of the tags sum and amount encodes the price. The integration system has to recognize synonymous tag names or translate the names between several languages.

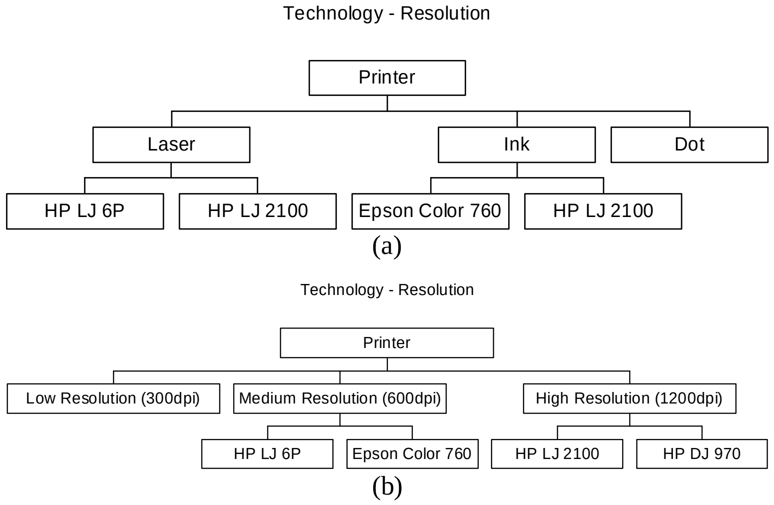

Fig. 4. Different product classifications

Finally, the descriptions can differ semantically. For example, two companies can use two different classifications of their products. The first company classifies the printers by printing technology into Laser, Ink, and Dot printers, as illustrated in Fig. 4(a). The second company classifies the printers by the resolution into Low Resolution, Medium Resolution and High Resolution printers (see Fig. 4(b)).

In the catalog presented in Fig. 4(a) HP LaserJet 2100 printer appears as an instance of a Laser printer, while in the second catalog (Fig. 4(b)) it appears as a High Resolution printer together with some ink high-resolution printers. These differences are very important for (automated) product classification and development of the integration rules. However, they show themselves only as small differences in the XML product encoding (see Fig. 5(a) and (b), that (b) corresponds to Fig. 4). These 'hidden' differences provide several additional problems in creation of the integration rules.

```
<printer>
<name>HP LJ 2100</name>
<subclassof>laser_printers</subclassof>
</printer>
```
(a)

```
<printer>
<name>HP LJ 2100</name>
<subclassof>1200dpi_printers</subclassof>
</printer>
```
(b)

Fig. 5. Semantic differences in different representations

3. THE REQUIREMENTS FOR B2B ONTOLOGY INTEGRATION

In the B2B area the number of product catalogs is relatively small compared to the B2C (Business-to-Customer) or B2B (Business-to-Business) areas, but each catalog contains a huge number of product descriptions. Once created and verified, a rule will be applied many times to a large number of product descriptions. Often these rules are constructed manually and they must be carefully verified from both business and technical points of view. This gives us the first requirement for ontology integration:

(1) The rules must be understandable by a domain specialist, who may not be a technical expert.

B2B suppliers provide their catalogs in a syntactically unified way, where XML becomes a de-facto standard, and several standards for product descriptions have already been proposed [4]. Consequently,

(2) The rules must be able to translate XML representations of product catalogs.

B2B participants tend to sell complicated products, and the corresponding product ontologies become very complicated. The integration on the instance level becomes not sufficient any more, and requires an additional schema-level integration:

(3) The rules must be able to deal with ontology schemas: classes, attributes, inheritance, etc.

Electronic product catalogs are used not only for trade mediation, they also can improve the supply chain used by a company. E-catalogs can support information flows in the supply chain used by the company in its interactions with suppliers and customers, and facilitate B2B procurement [1]. This gives the following requirements for the integration:

(4) It must perform integration compatible with the representations used by the legacy systems, which usually have a flat structure and are based on the database technology;
(5) The integration must be compatible with the security services used in the company.

Each product catalog is viewed and maintained by a number of people from different departments, who have different functions and need different *views* on the catalog. The same problem appears in the area of web-site development for e-business [9], that requires generation of a separate view for each customer. This requires the integration process to be able to:

(6) Automatically translate complicated product descriptions into their light-weight representations, or *views*.
(7) Be easy adaptable to changes in the view profiles, or in these light-weight product description formats.

Requirement (1) implicitly points to a special view on the integration rules, used by the people who develop and validate them.

Huge catalog size and a large number of the users inspire development of intelligent sales assistants for the Web that consult and guide the customers between product catalogs [10]. These services bring the following requirements:

(8) The ontologies must be standardized on the syntactical level with XML, that is already a de-facto standard.
(9) The ontologies must be 'derivable' from different views, as a reverse to requirement (6). For example, a product description can be derived form its technical and market descriptions.
(10) The tool must be able to integrate specific product ontologies with general domain ontologies.

Product after-sales support and maintenance [5] requires integration of complicated product ontologies, where the individuals have a rich structure, which may not strictly correspond to the ontology. For example, each crane can have its own particular configuration, opposite to an office printer that has the same configuration as many other printers of the same model. This area gives the following:

(11) Ontology instances that describe the final products may differ from the product ontology; besides ontology integration, product integration will require non-trivial integration of ontology instances that may not strictly correspond to the ontologies.

The existing semantical-level integration approaches (i.e. [3]) define the integration rules manually based on the underlying product models. Despite of improved quality of the rules, this ignores requirements (1), (6), (7), and especially (11).

Syntactical integration defines the rules in the terms of the class and attribute names used in the ontologies to be integrated. The integration rules are conceptually blind but they are relatively easy to develop and implement. This level is widely used in the database community for database schema integration [2], e-commerce [7], and helps to satisfy requirements (4)-(7).

4. EXISTENT ONTOLOGY INTEGRATION TOOLS

There are two ontology integration tools developed in the knowledge engineering community: Chimaera [6] and PROMPT [8]. Both tools support merging of ontological terms (class and attribute names) from varied sources. During the class merging process they present to the user the pairs of classes with similar names that either represent the same class from the input ontologies, or might require some taxonomic edition to make one a subclass of the other. Then a human user decides which integration operation to apply to the pair of classes, and the system guides him to the next pair. PROMPT provides more automation in ontology merging than Chimaera. After the user makes the integration decision, PROMPT suggests to perform a sequence of actions on copying the classes and their attributes, creating necessary subclasses and putting them in the right places in the hierarchy. The sequences are hard-encoded into the system, but experiments have shown that they perform very well. In both approaches the user still has to decide which ontology integration operation to apply for each pair of classes.

These tools satisfy requirements (3)-(5) and (10), and they can be easily extended with XML compatibility to satisfy requirements (2) and (8). Both tools fail to perform automated integration as required in (6), (7), and it is unclear how far they correspond to requirements (9) and (11).

5. CONCLUSIONS

The requirements collected in the paper show that B2B product integration system requires two important parts. First, it might contain an integration kernel of the rules, which are manually verified, and actually translate the product descriptions in the XML format. This requires the tools that assist the user in the rule creation and validation process. Improving the development of the kernel integration rules is crucially important now for B2B trade mediation.

Second, it might be able to generate numerous views on the catalog, according to the requirements of various customers and departments. Some applications require a reverse operation, in which the product description is created from various views. In the near future, the companies will start to automate their supply chain and will require efficient view processing and product ontology integration. Special research is needed to develop the methods for (semi)automatic creation of the views, and for constructing a complete product descriptions out of them.

ACKNOWLEDGEMENTS

The author would like to thank Dieter Fensel for helpful discussions and comments on this paper.

REFERENCES

[1] Baron, J., Shaw, M., Bailey, A.: Web-based E-catalog Systems in B2B Procurement. Communications of the ACM **43**:5 (2000) 93-100

[2] Batini, C., Lenzerini, M., Navathe, S.: A comparative analysis of methodologies for database schema integration. ACM Computing Surveys **18**:4 (1986) 323-364

[3] Bowers, S., Delcambre, L.: Representing and Transforming Model-Based Information. In: Proceedings of the Workshop on the Semantic Web at ECDL-00, Lisbon, Portugal, September 21 (2000)

[4] Li, H.: XML and Industrial Standards for Electronic Commerce. Knowledge and Information Systems **2**:4 (2000) 487-497

[5] Mannisto, T., Peltonen, H., Soininen, T., Sulonen, R.: Multiple Abstraction Levels In Modelling Product Structures. Data & Knowledge Engineering **36** (2001) 55-78

[6] McGuinness, D., Fikes, R., Rice, J., Wilder, S.: An Environment for Merging and Testing Large Ontologies. In: Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000), Breckenridge, Colorado, April 12 **-** 15 (2000)

[7] Ng, W., Yan, G., Lim, E.: Heterogeneous Product Description in Electronic Commerce. SIGecom Exchanges, Newsletter of the ACM Special Interest Group on E-commerce **1** (2000)

[8] Noy, N., Musen, M.: PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. In: Proceedings of the AAAI-00 Conference, Austin, TX (2000)

[9] Santesmases, J., Lopez, D., Fernandez, Y.: SPGC: A Personalized Server and Content Manager. In: E-business: Issues, Applications and Developments. Proceedings of eBusiness and eWork 2000 Conference and Exhibition, Madrid, Spain, 18-20 October (2000)

[10] Traphoner, R.: WEBSELL: Intelligent Sales Assistants for the World Wide Web. In: E-business: Issues, Applications and Developments. Proceedings of eBusiness and eWork 2000 Conference and Exhibition, Madrid, Spain, 18-20 October (2000)